Personalizar e usar o ambiente shell

www.4linux.com.br

Sumário

Índice de tabelas

Índice de Figuras

Capítulo 1

Personalizar e usar o ambiente shell

1.1. Objetivos

- Definir as variáveis de ambiente no início da sessão;
- Funções BASH para sequências de comandos frequentemente utilizados;
- Definir o caminho de pesquisa de comando com o diretório apropriado.

1.2. Mãos a obra

Em distribuições GNU/Linux o uso de variáveis permite ao administrador do sistema organizar informações e possibilita a personalização de valores e informações aos usuários.

O que é uma variável?

São espaços na memória que armazenam valores.

Variáveis locais x globais

Quando você declara uma variável ela pode ser considerada pelo sistema como local ou global. A diferença está na maneira de declarar a variável. Exemplo:

Variável local

$ curso=4linux

Variável global

$ export curso=4linux

Veja que o comando export foi usado antes de declarar a variável. Para listar os tipos de variáveis use os comandos env e set:

Para exibir variáveis locais use o comando set

$ set

Para exibir variáveis globais use o comando env

$ env

Para deletar uma variável da memória use o comando unset

$ unset curso

Variáveis de ambiente

O shell guarda informações do ambiente dentro de algumas variáveis, chamadas de variáveis de ambiente. Veja a descrição abaixo:

HOME – Exibe o diretório do usuário logado;

Exemplo:

$ echo $HOME

```
/home/aluno
```

SHELL – Exibe qual shell está sendo usado;

Exemplo:

$ echo $SHELL

```
/bin/bash
```

TERM – Exibe o tipo de terminal que está sendo usado;

Exemplo:

$ echo $TERM

```
linux
```

USER – Exibe o nome do usuário logado;

Exemplo:

$ echo $USER

```
aluno
```

PATH – Exibe quais diretórios pesquisar e a ordem na qual eles são pesquisados para encontrar um determinado comando;

Exemplo:

$ echo $PATH

```
/usr/local/bin:/usr/bin:/bin:/usr/games
```

MAIL – Exibe o local onde ficam armazenados os emails do usuário logado;

Exemplo:

$ echo $MAIL

```
/var/mail/aluno
```

OSTYPE – Exibe o tipo de sistema operacional em uso;

Exemplo:

$ echo $OSTYPE

```
linux-gnu
```

HISTSIZE – Exibe a quantidade de comandos que o histórico do Linux pode armazenar;

Exemplo:

$ echo $HISTSIZE

```
500
```

TMOUT – Exibe a quantidade de tempo (em segundos) que o bash vai encerrar a sessão do usuário atual, com base em sua inatividade no sistema;

Exemplo:

$ echo $TMOUT

```
60
```

PWD – Exibe a localização do diretório atual;

Exemplo:

$ echo $PWD

```
/home/aluno
```

OLDPWD – Exibe a localização do diretório anterior;

Exemplo:

$ echo $OLDPWD

```
/tmp
```

PS1 – Exibe a aparência do prompt, como o nome de usuário, maquina e diretório atual;

Exemplo:

$ echo $PS1

```
${debian_chroot:+($debian_chroot)}\u@\h:\w\$
```

Não esqueça que os nome das variáveis de ambiente são apresentadas em maiúsculas!

Apelidos de comandos

Para facilitar a administração no sistema é possível usar nomes curtos aos comandos que se tornam extensos, devido a quantidade de opções usadas como flags.

Exemplo de uso do comando ls:

$ ls –color -lah /etc

No exemplo a opção –color foi utilizada para cor, -l para detalhes, -a para arquivos ocultos e h exibe os tamanhos em formato mais humano (10K, 10M e 10G)

Para facilitar o uso do comando ls e todas essas opções, você criar um alias (apelido) através do comando alias. Vamos a prática:

$ alias ll=”ls –color -lah“

Para verificar se o apelido foi criado, use sem nenhuma opção o comando alias

$ alias

```
alias ll='ls --color -lah'
```

Agora toda vez que precisar usar o ls com as opções, use o alias ll

$ ll

Para apagar o alias criado use o comando unalias e nome do apelido

$ unalias ll

Capítulo 2

Gerenciando

2.1. Objetivos

- Trobleshooting: Trabalhar com os arquivos: ~ /. bash_profile, ~ /. bash_login, ~ /. profile, ~ /. bashrc e ~ /. bash_logout.

2.2. Troubleshooting

Como posso personalizar meu ambiente shell?

Cada usuário pode personalizar seu ambiente através do shell, declarando variáveis, apelidos para comandos (alias) e ainda executar comandos ou scripts no login e logout. Veja a lista de arquivos que podem ser personalizados.

/etc/profile – Este arquivo contém comandos que são executados para todos os usuários do sistema no momento do login (somente o usuário pode editar);

~/.bash_profile – Executado por shells que usam autenticação (nome e senha). Para root o arquivo é o .profile;

~/.bashrc – Executado por shells que não requerem autenticação (seção de terminal no X);

~/.bash_logout – É lido e executado toda vez que saímos de um shell;

~/.bash_history – Lista dos comandos digitados pelos usuários.

Como exemplo pratico vamos personalizar o login de um usuário para exibir um calendário, alem de declarar uma variável e criar um alias para um comando.

Com um usuário comum faça login no sistema e abra o arquivo .bashrc

$ vim .bashrc

```
 1 # ~/.bashrc: executed by bash(1) for non-login shells.
 2 # see /usr/share/doc/bash/examples/startup-files (in the package bash-doc)
 3 # for examples
 4
 5 # If not running interactively, don't do anything
 6 [ -z "$PS1" ] && return
 7
 8 # don't put duplicate lines in the history. See bash(1) for more options
 9 # don't overwrite GNU Midnight Commander's setting of `ignorespace'.
10 export HISTCONTROL=$HISTCONTROL${HISTCONTROL+,}ignoredups
11 # ... or force ignoredups and ignorespace
12 export HISTCONTROL=ignoreboth
13
14 # append to the history file, don't overwrite it
15 shopt -s histappend
16
17 # for setting history length see HISTSIZE and HISTFILESIZE in bash(1)
18
19 # check the window size after each command and, if necessary,
20 # update the values of LINES and COLUMNS.
21 shopt -s checkwinsize
22
23 # make less more friendly for non-text input files, see lesspipe(1)
24 #[ -x /usr/bin/lesspipe ] && eval "$(SHELL=/bin/sh lesspipe)"
```

Na linha 4 adicione um comando, linha 5 um alias e na linha 6 uma variável

```
 1 # ~/.bashrc: executed by bash(1) for non-login shells.
 2 # see /usr/share/doc/bash/examples/startup-files (in the package bash-doc)
 3 # for examples
 4 cal
 5 alias ll="ls -lah --color"
 6 curso=4linux
 7 # If not running interactively, don't do anything
 8 [ -z "$PS1" ] && return
```

Faça logout com o usuário e ao se logar teste o alias e a variável.

```
Debian GNU/Linux 5.0 maq2 tty1

maq2 login: aluno
Password: _
```

$ ll

```
drwxr-xr-x 14 aluno aluno 4,0K Set  1 19:01 .
drwxr-xr-x  7 root  root  4,0K Set  1 20:43 ..
-rw-------  1 aluno aluno  374 Set  1 19:23 .bash_history
-rw-r--r--  1 aluno aluno   15 Ago 21 00:19 .bash_logout
-rw-r--r--  1 aluno aluno 3,1K Ago 21 00:20 .bashrc
drwx------  3 aluno aluno 4,0K Fev 24  2010 .dbus
drwxr-xr-x  2 aluno aluno 4,0K Fev 24  2010 Desktop
-rw-------  1 aluno aluno   28 Fev 24  2010 .dmrc
-rw-r--r--  1 aluno aluno  786 Ago 20 15:42 fstab
drwx------  4 aluno aluno 4,0K Fev 24  2010 .gconf
drwx------  2 aluno aluno 4,0K Fev 24  2010 .gconfd
drwxr-xr-x  3 aluno aluno 4,0K Fev 24  2010 .gnome
drwx------  7 aluno aluno 4,0K Fev 24  2010 .gnome2
drwx------  2 aluno aluno 4,0K Fev 24  2010 .gnome2_private
drwx------  2 aluno aluno 4,0K Fev 24  2010 .gnupg
drwxr-xr-x  2 aluno aluno 4,0K Fev 24  2010 .gstreamer-0.10
-rw-------  1 aluno aluno  159 Fev 24  2010 .ICEauthority
-rw-r--r--  1 aluno aluno   12 Ago 20 15:42 .message
drwx------  3 aluno aluno 4,0K Fev 24  2010 .metacity
drwxr-xr-x  3 aluno aluno 4,0K Fev 24  2010 .nautilus
-rw-r--r--  1 aluno aluno  675 Fev 24  2010 .profile
drwx------  2 aluno aluno 4,0K Fev 24  2010 .ssh
-rw-------  1 aluno aluno  665 Set  1 19:01 .viminfo
-rw-r--r--  1 aluno aluno  818 Fev 24  2010 .xsession-errors
```

$ echo $curso